

TERMINAL RELIABILITY OF ROAD NETWORKS WITH MULTIPLE DESTINATION OPTIONS

THORSTEN NEUMANN* & MICHAEL BEHRISCH**

German Aerospace Center (DLR), Institute of Transportation Systems, Berlin, Germany.

ABSTRACT

In crisis situations such as natural disasters, it is essential that people can be reached by fire and rescue services as well as police forces. Moreover, access to food and water supply should be ensured for everyone, for instance. In all cases, people depend on the (potentially damaged) road infrastructure – modelled as a graph network – that for all nodes in the network should provide connections to at least one supply node of each type (i.e., fire department, hospital etc.) even if some links in the network are currently unavailable. Assuming that there are multiple supply nodes per type among the whole network, the present contribution discusses the risk that a given node of the network becomes isolated from all of these supply nodes depending on the topological structure of the network. For this purpose, the well-known concept of terminal reliability is adapted to the situation with multiple possible destinations which is realized by appropriately modifying the original graph. An algorithm is presented that allows finding all relevant cut sets in the modified graph which can be used for computing the probability R_{sys} that a given node remains connected to at least one of the supply nodes considered depending on the link failure probabilities p_i . A simple clustering approach together with Boolean algebra finally yields explicit numbers for R_{sys} depending on p_i . The whole concept is demonstrated based on an illustrative example showing the different endangerment among the nodes of the network including a discussion about the specific criticality of each network link with regard to ensuring the connectivity between supply nodes and other nodes. Thus, by identifying critical links and quantifying the risk of isolation for all nodes in the network, the present contribution provides useful tools for prevention in crisis management.

Keywords: resilience, vulnerability, reliability analysis, road networks, minimal cut sets, algorithm

1 INTRODUCTION

The analysis of resilience and vulnerability of critical infrastructures is an essential aspect of crisis prevention and management (cf. [1]) and is still under research (cf. [2]). The Transportation Research Board (TRB) of the National Academies of Science – Engineering – Medicine in the United States, for example, periodically publishes a detailed overview of their corresponding activities (cf. [3]). In this broad context, the present contribution particularly discusses a method and results with regard to the probabilities that certain nodes of a road network become isolated from a given subset of other nodes which are considered to provide specific – potentially life-critical – services. These “supply nodes” could be fire departments, hospitals or also supermarkets, for instance, and ideally need to be reached even if several links of the network are unavailable because of natural disasters or other catastrophic events.

The main questions in the following are: (1) How likely is it that a given node of the network loses its connection to all supply nodes of a selected type (i.e., fire department, hospital etc.) depending on network topology and link failure probabilities? (2) What are the most critical links in the network in terms of negatively affecting the isolation risk of many nodes when becoming unavailable?

* ORCID: <http://orcid.org/0000-0002-9236-0585>

** ORCID: <http://orcid.org/0000-0002-0032-7930>

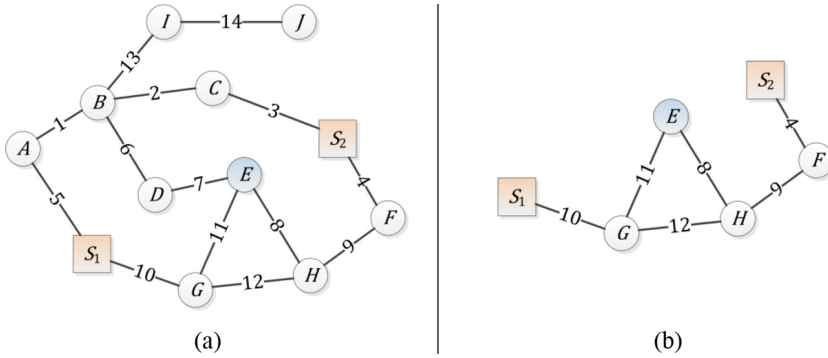


Figure 1: Exemplary road network with two supply nodes of the same type: (a) complete network; (b) reduced network.

For illustrative purposes, we will consider the artificial road network from Fig. 1a or a specific subset of it as in Fig. 1b. The whole network consists of ten regular nodes ($A-J$) which are connected with two distinguished supply nodes of the same type (S_1 and S_2) via fourteen bidirectional links (1–14). In some cases, we will focus specifically on the exemplary node E in the center of the network for demonstrating our approach. Finally, let $p_i \in [0,1]$ be the link failure probabilities for $i=1,\dots,14$ and I_i the corresponding indicator functions representing the availability of the links, i.e.

$$I_i = \begin{cases} 1 & \text{if link } i \text{ is available,} \\ 0 & \text{if link } i \text{ is not available.} \end{cases} \quad (1)$$

For simplicity, we further assume that all I_i are stochastically independent even if this could be unrealistic in some cases (e.g., flooding).

2 METHODOLOGY

Answering the questions from above (see Section 1) is strongly related to the concept of terminal reliability in graph networks (cf. [4]). Given a specific origin-destination pair (X,Y) in the network, this addresses the probability that the corresponding nodes remain connected depending on potential link failures. Often, the terms *connectivity reliability* and *terminal reliability* are considered to be the same in the literature (cf. [1]). However note that connectivity of a graph usually refers to the situation that there are paths not only between two particular nodes but between all nodes of the network which definitely is something different.

Let now $P(k)$ for $k=1,\dots,r$ be the (minimal) paths between two selected nodes X and Y of a given network with (stochastically independent) link indicator functions I_i as in Eq. (1) for all links of the network ($i=1,\dots,m$), where (*minimal*) *path* – in contrast to the more general term *way* – means that $P(k)$ does not connect X and Y anymore whenever any of the elements of $P(k)$ is omitted. That is, the set $P(k)$ consists of all link indices of the k -th path between X and Y in this case. It can be shown then (cf. [4]) that the so-called structure function

$$\psi(I_1, \dots, I_m) := 1 - \prod_{k=1}^r \left(1 - \prod_{i \in P(k)} I_i\right) \quad (2)$$

represents whether there is a connection between X and Y available or not, i.e.

$$\psi(I_1, \dots, I_m) = \begin{cases} 1 & \text{if there is at least one connection between } X \text{ and } Y \text{ available.} \\ 0 & \text{no} \end{cases} \quad (3)$$

Alternatively, $\psi(I_1, \dots, I_m)$ can also be expressed based on the so-called (minimal) cut sets belonging to the origin-destination pair (X, Y) . Here, a cut set Z is a set of link indices such that X and Y lose their connection if all corresponding links in the network fail. Consequently, $Z := \{1, \dots, m\}$ is a trivial cut set as it simply disconnects all nodes of the considered network. On the contrary, a cut set Z is called *minimal* if Z is no cut set anymore whenever any element of Z is eliminated from it. Let now $Z(k)$ for $k = 1, \dots, r$ be the minimal cut sets related to the connection between X and Y , then

$$\psi(I_1, \dots, I_m) = \prod_{k=1}^{r'} \left(1 - \prod_{i \in Z(k)} (1 - I_i)\right). \quad (4)$$

However, note that finding all these cut sets manually can be laborious, even if the considered networks are small. Thus, Section 2.3 will provide a recursive algorithm that generates all minimal cut sets based on the paths between two given nodes.

We now ask for the probability that X and Y become disconnected depending on given link failure probabilities p_i for $i = 1, \dots, m$ (cf. Section 1). Obviously, I_i for $i = 1, \dots, m$ has a Bernoulli distribution with parameter $R_i := 1 - p_i$ which is called *link reliability* in this context. Moreover, $\psi(I_1, \dots, I_m)$ is Bernoulli-distributed as well, and the so-called system reliability R_{sys} (i.e., the probability of having at least one available connection between X and Y) is given by

$$R_{\text{sys}} = \mathbb{P}(\psi(I_1, \dots, I_m) = 1) = \mathbb{E}(\psi(I_1, \dots, I_m)). \quad (5)$$

As a short example, we consider the (reduced) network from Fig. 1b and ask for the reliability of the connection between the nodes E and F given the corresponding link failure probabilities p_i . We obtain the link reliabilities $R_i = 1 - p_i$ for all i , and Eq. (2) yields the relevant structure function ψ based on the two existing paths $P(1) = \{8, 9\}$ and $P(2) = \{9, 11, 12\}$ between E and F , namely – note that $(I_i)^b = I_i$ for all $b \in \mathbb{N}$ –

$$\psi(I_4, I_8, I_9, I_{10}, I_{11}, I_{12}) = 1 - (1 - I_8 I_9)(1 - I_9 I_{11} I_{12}) = I_8 I_9 + I_9 I_{11} I_{12} - I_8 I_9 I_{11} I_{12}. \quad (6)$$

Consequently,

$$R_{\text{sys}} = \mathbb{E}(\psi(I_4, I_8, I_9, I_{10}, I_{11}, I_{12})) = R_8 R_9 + R_9 R_{11} R_{12} - R_8 R_9 R_{11} R_{12} \quad (7)$$

in this example due to the fact that the appearing indicator functions I_i were considered to be stochastically independent (see Section 1).

2.1 Graph modification and path search

Terminal reliability as explained in the previous section refers to the connection between two given nodes of a network, i.e. exactly one origin and one destination node. On the contrary, the actual focus of the present study is on the situation with multiple destinations where it is sufficient that the given origin remains connected to at least one of these possible destinations

(cf. Section 1). In case of an arbitrary regular node from the network in Fig. 1a, for instance, we require that one of the supply nodes can be reached. It is not mandatory that connections to both supply nodes are available at the same time.

Unfortunately, the corresponding system reliability R_{sys} (i.e., the probability that there is at least one available path to at least one of the supply nodes) generally is no simple function of the separate probabilities that either the first or the second supply node can be reached. This is because there may be (partly) overlapping paths with regard to both these cases causing complex (stochastic) dependencies in consequence of the explicit network topology. Hence, we cannot apply the previously described concept of terminal reliability directly at this point. However, there is a simple solution.

For this purpose, consider the modified graph networks from Fig. 2 where all supply nodes (of the same type) with their adjacent links were combined to form a single “virtual” supply node S (which can be considered as being located in a third dimension above the rest of the network, for instance). For a given regular node X of the original network from Fig. 1, the availability of a connection to at least one of the original supply nodes is equivalent then to the availability of a path between X and S in the modified graph. Thus, by passing over to the modified network, terminal reliability becomes applicable again for answering our original question.

In case of node E in the complete network from Fig. 1a, for instance, the relevant paths between E and S in the modified graph (see Fig. 2a) are given by $P(1) = \{1, 5, 6, 7\}$, $P(2) = \{2, 3, 6, 7\}$, $P(3) = \{10, 11\}$, $P(4) = \{8, 10, 12\}$, $P(5) = \{4, 8, 9\}$ and $P(6) = \{4, 9, 11, 12\}$. Based on Eqs. (2) and (5), it is possible then to compute the probability R_{sys} that E remains connected to at least one of the supply nodes. Needless to say, the probability that E becomes isolated from both supply nodes simply is $Q_{\text{sys}} := 1 - R_{\text{sys}}$. Finding all paths between two given nodes of a network can be done by recursively enumerating all possible neighbors using standard depth-first search algorithms (cf. [5]) although, of course, there are well-known practical limitations due to the fact that the number of paths may grow exponentially with the number of nodes in the network.

2.2 Clustering and reliability estimation

Usually, the full expansion of the structure function ψ as in Eq. (6) can be very cumbersome to obtain (cf. [6]). Consequently, several exact and heuristic approaches have been developed

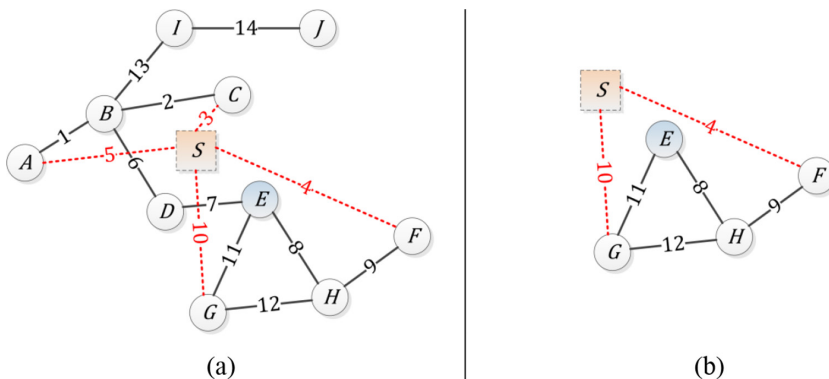


Figure 2: Modified exemplary network with a “virtual” supply node: (a) complete network; (b) reduced network.

to optimize this computation (cf. [4]). But even in the standard case using Boolean algebra as above, it is sometimes possible to reduce the effort by two simple tricks (cf. Lemma 1 and 2).

Lemma 1: Let $P(k)$ for $k = 1, \dots, r$ be the (minimal) paths between two selected nodes of a given network with (stochastically independent) link indicator functions I_i and link reliabilities R_i for $i = 1, \dots, m$. Moreover, let $i_0 \in \{1, \dots, m\}$ be such that $i_0 \in P(k)$ for all k . Then,

$$R_{\text{sys}} = \mathbb{E}(\psi(I_1, \dots, I_m)) = R_{i_0} \cdot \mathbb{E}\left(1 - \prod_{k=1}^r \left(1 - \prod_{i \in P(k) \setminus \{i_0\}} I_i\right)\right). \quad (8)$$

In other words, the indicator function I_{i_0} can be excluded from the detailed expansion of ψ when computing its expectation if *all* paths contain the link i_0 . In case of the network from Fig. 1a, this becomes relevant for the nodes I and J , for instance, when considering their connection to the given supply nodes.

Proof of Lemma 1: We prove the proposition of Lemma 1 by mathematical induction. Thus, let $r = 1$ first. Due to the stochastic independence of the indicator functions I_i , we obtain (cf. Eqs. (2) and (5))

$$R_{\text{sys}} = \mathbb{E}\left(\prod_{i \in P(1)} I_i\right) = \mathbb{E}(I_{i_0}) \cdot \mathbb{E}\left(\prod_{i \in P(1) \setminus \{i_0\}} I_i\right) = R_{i_0} \cdot \mathbb{E}\left(1 - \left(1 - \prod_{i \in P(1) \setminus \{i_0\}} I_i\right)\right).$$

We now show that Eq. (8) also holds for $r + 1$ given that it is correct for any given $r \in \mathbb{N}$. Namely, having in mind that $(I_{i_0})^b = I_{i_0}$ for all $b \in \mathbb{N}$ and that $i_0 \in P(k)$ for all k , the following holds:

$$\begin{aligned} R_{\text{sys}} &= \mathbb{E}\left(1 - \prod_{k=1}^{r+1} \left(1 - \prod_{i \in P(k)} I_i\right)\right) = \mathbb{E}\left(1 - \left(1 - \prod_{i \in P(r+1)} I_i\right) \prod_{k=1}^r \left(1 - \prod_{i \in P(k)} I_i\right)\right) \\ &= \mathbb{E}\left(1 - \prod_{k=1}^r \left(1 - \prod_{i \in P(k)} I_i\right)\right) + \mathbb{E}\left(\prod_{i \in P(r+1)} I_i \cdot \prod_{k=1}^r \left(1 - \prod_{i \in P(k)} I_i\right)\right) \\ &= R_{i_0} \mathbb{E}\left(1 - \prod_{k=1}^r \left(1 - \prod_{i \in P(k) \setminus \{i_0\}} I_i\right)\right) + \mathbb{E}\left(I_{i_0} \prod_{i \in P(r+1) \setminus \{i_0\}} I_i \prod_{k=1}^r \left(1 - \prod_{i \in P(k)} I_i\right)\right) \\ &= R_{i_0} \mathbb{E}\left(1 - \prod_{k=1}^r \left(1 - \prod_{i \in P(k) \setminus \{i_0\}} I_i\right)\right) + \mathbb{E}\left(\prod_{i \in P(r+1) \setminus \{i_0\}} I_i \prod_{k=1}^r \left(I_{i_0} - \prod_{i \in P(k)} I_i\right)\right) \\ &= R_{i_0} \mathbb{E}\left(1 - \prod_{k=1}^r \left(1 - \prod_{i \in P(k) \setminus \{i_0\}} I_i\right)\right) + \mathbb{E}\left(I_{i_0} \prod_{i \in P(r+1) \setminus \{i_0\}} I_i \prod_{k=1}^r \left(1 - \prod_{i \in P(k) \setminus \{i_0\}} I_i\right)\right) \\ &= R_{i_0} \mathbb{E}\left(1 - \prod_{k=1}^r \left(1 - \prod_{i \in P(k) \setminus \{i_0\}} I_i\right)\right) + R_{i_0} \mathbb{E}\left(\prod_{i \in P(r+1) \setminus \{i_0\}} I_i \prod_{k=1}^r \left(1 - \prod_{i \in P(k) \setminus \{i_0\}} I_i\right)\right) \\ &= R_{i_0} \mathbb{E}\left(1 - \left(1 - \prod_{i \in P(r+1) \setminus \{i_0\}} I_i\right) \prod_{k=1}^r \left(1 - \prod_{i \in P(k) \setminus \{i_0\}} I_i\right)\right) \\ &= R_{i_0} \mathbb{E}\left(1 - \prod_{k=1}^{r+1} \left(1 - \prod_{i \in P(k) \setminus \{i_0\}} I_i\right)\right). \end{aligned}$$

Lemma 2: Let $P(k)$ for $k = 1, \dots, r$ be the (minimal) paths between two selected nodes of a given network with (stochastically independent) link indicator functions I_i and link reliabilities R_i for $i = 1, \dots, m$. Moreover, let C_j for $j = 1, \dots, l$ be disjoint subsets of $\{1, \dots, m\}$ such that for each $k \in \{1, \dots, r\}$ there is a $j \in \{1, \dots, l\}$ with $P(k) \subseteq C_j$. Then,

$$R_{\text{sys}} = \mathbb{E}(\psi(I_1, \dots, I_m)) = 1 - \prod_{j=1}^l (1 - R_{\text{sys}}^{(j)}) \quad (9)$$

where

$$R_{\text{sys}}^{(j)} := \mathbb{E}\left(1 - \prod_{k:P(k) \subseteq C_j} \left(1 - \prod_{i \in P(k)} I_i\right)\right). \quad (10)$$

That means, if it is possible to cluster the paths $P(k)$ in such a way that there is no link overlap between any two paths from different clusters, the structure function ψ needs to be expanded per cluster only for computing the respective system reliability R_{sys} . Moreover, note that potentially the calculations can be simplified even further by applying Lemma 1 to the terms $R_{\text{sys}}^{(j)}$ from Eq. (10) whenever possible.

Proof of Lemma 2: Due to the stochastic independence of the indicator functions I_i and because of the fact that the clusters C_j are disjoint per assumption, we obtain

$$\begin{aligned} R_{\text{sys}} &= \mathbb{E}\left(1 - \prod_{k=1}^l \left(1 - \prod_{i \in P(k)} I_i\right)\right) = 1 - \mathbb{E}\left(\prod_{j=1}^l \prod_{k:P(k) \subseteq C_j} \left(1 - \prod_{i \in P(k)} I_i\right)\right) \\ &= 1 - \prod_{j=1}^l \mathbb{E}\left(\prod_{k:P(k) \subseteq C_j} \left(1 - \prod_{i \in P(k)} I_i\right)\right). \end{aligned}$$

We demonstrate the above clustering approach in combination with Lemma 1 in case of node E from the network in Fig. 1a considering its connection to at least one of the supply nodes. The relevant paths have already been found (see Section 2.1) and are defined by $P(1) = \{1, 5, 6, 7\}$, $P(2) = \{2, 3, 6, 7\}$, $P(3) = \{10, 11\}$, $P(4) = \{8, 10, 12\}$, $P(5) = \{4, 8, 9\}$ and $P(6) = \{4, 9, 11, 12\}$. Obviously, there are two natural clusters. Namely, $C_1 = \{1, 2, 3, 5, 6, 7\}$ and $C_2 = \{4, 8, 9, 10, 11, 12\}$ with $P(1)$ and $P(2)$ belonging to cluster C_1 and $P(3)$, $P(4)$, $P(5)$ as well as $P(6)$ belonging to cluster C_2 . Moreover, the links 6 and 7 are part of all paths in cluster C_1 while the cluster C_2 comprises paths that are even completely disjoint (i.e., $P(3)$ and $P(5)$). According to Lemma 1 and 2, the system reliability R_{sys} is thus given by

$$R_{\text{sys}} = 1 - \left(1 - R_{\text{sys}}^{(1)}\right) \left(1 - R_{\text{sys}}^{(2)}\right) \quad (11)$$

where

$$\begin{aligned} R_{\text{sys}}^{(1)} &= R_6 R_7 \mathbb{E}\left(1 - (1 - I_1 I_5)(1 - I_2 I_3)\right) = R_6 R_7 \mathbb{E}(I_2 I_3 + I_1 I_5 - I_1 I_2 I_3 I_5) \\ &= R_6 R_7 (R_2 R_3 + R_1 R_5 - R_1 R_2 R_3 R_5) \end{aligned} \quad (12)$$

and

$$\begin{aligned} R_{\text{sys}}^{(2)} &= \mathbb{E}\left(1 - (1 - I_{10} I_{11})(1 - I_8 I_{10} I_{12})(1 - I_4 I_8 I_9)(1 - I_4 I_9 I_{11} I_{12})\right) \\ &= \mathbb{E}\left(I_{10} I_{11} + I_4 I_8 I_9 + I_8 I_{10} I_{12} + I_4 I_9 I_{11} I_{12} - I_8 I_{10} I_{11} I_{12} - I_4 I_8 I_9 I_{10} I_{11} - \dots \right. \\ &\quad \left. \dots - I_4 I_8 I_9 I_{10} I_{12} - I_4 I_8 I_9 I_{11} I_{12} - I_4 I_9 I_{10} I_{11} I_{12} + 2 I_4 I_8 I_9 I_{10} I_{11} I_{12}\right) \end{aligned} \quad (13)$$

$$\begin{aligned}
&= R_{10}R_{11} + R_4R_8R_9 + R_8R_{10}R_{12} + R_4R_9R_{11}R_{12} - R_8R_{10}R_{11}R_{12} - R_4R_8R_9R_{10}R_{11} - \dots \\
&\dots - R_4R_8R_9R_{10}R_{12} - R_4R_8R_9R_{11}R_{12} - R_4R_9R_{10}R_{11}R_{12} + 2R_4R_8R_9R_{10}R_{11}R_{12}.
\end{aligned}$$

Assuming that $R_i \equiv R := 1 - p$ for all i where $p \in [0, 1]$ is a constant link failure probability for all links, we finally obtain

$$\begin{aligned}
R_{\text{sys}} &= 1 - (1 - 2R^4 + R^6)(1 - R^2 - 2R^3 + 4R^5 - 2R^6) \\
&= R^2 + 2R^3 + 2R^4 - 4R^5 - R^6 - 4R^7 + R^8 + 10R^9 - 4R^{10} - 4R^{11} + 2R^{12}.
\end{aligned} \tag{14}$$

Figure 4 in Section 3 shows (among others) a plot of this R_{sys} as a function of p .

2.3 Cut set detection and link criticality

As already discussed, the system reliability R_{sys} can also be expressed based on (minimal) cut sets (see Eq. (4)). Moreover, cut sets may be used to identify the most critical links in a network. For, whenever a cut set is active (i.e., when all links belonging to it fail), the corresponding structure function ψ from Eq. (3) becomes zero indicating that there is no available connection between the considered nodes (i.e., between regular nodes and supply nodes in our specific case). This means that links that are part of many cut sets (and/or particular small cut sets) can be seen to be more critical than others in the network.

Therefore, in order to answer our second question from Section 1, it is necessary to obtain all (minimal) cut sets between the regular nodes and supply nodes first. More precisely, we need to know the (minimal) cut sets between the regular nodes and the corresponding “virtual” supply node in the modified graph (cf. Fig. 2). Figure 3 presents the pseudo-code of a recursive algorithm that detects all (minimal) cut sets based on the knowledge about all paths between the nodes of a given origin-destination (OD) pair.

The basic idea of the algorithm from Fig. 3 is to systematically generate link combinations such that all relevant paths become cut when the respective links fail. Once a candidate for a cut set is found, it is checked with regard to minimality. We demonstrate the algorithm’s functionality using the node E from the reduced network in Fig. 1b as an example. In this context, we ask for connectivity to at least one of the supply nodes. Consequently (cf. Section 2.1), the relevant paths are given by $P(1) = \{10, 11\}$, $P(2) = \{8, 10, 12\}$, $P(3) = \{4, 8, 9\}$ and $P(4) = \{4, 9, 11, 12\}$.

Obviously, in order to cut $P(1)$, either link 10 or link 11 must fail. We start with the case that link 10 becomes unavailable. Then, $P(2)$ is disconnected directly as well. Thus, we can pass over to $P(3)$. Given that link 4 becomes interrupted, we obtain that then $P(4)$ is cut, too. Hence, as there are no more paths, $\tilde{Z}(1) := \{4, 10\}$ is the first candidate for a minimal cut set. We then move back to $P(3)$ and check what happens if link 8 fails instead of link 4. Obviously, now $P(4)$ remains connected so that link 4, 9, 11 or 12 needs to be interrupted additionally in order to get a candidate cut set. Obviously, the so-found cut set $\tilde{Z}(2) := \{4, 8, 10\}$ is not minimal because of $\tilde{Z}(2) \supset \tilde{Z}(1)$. On the other hand, $\tilde{Z}(3) := \{8, 9, 10\}$, $\tilde{Z}(4) := \{8, 10, 11\}$ and $\tilde{Z}(5) := \{8, 10, 12\}$ will be added to the candidate list (at least for the moment). Again, we go back to $P(3)$ with the assumption that (as last option) link 9 fails. Clearly, $P(4)$ is not available then anymore, and $\tilde{Z}(6) := \{9, 10\}$ becomes another candidate cut set directly. Furthermore, $\tilde{Z}(6) \subset \tilde{Z}(3)$ so that ultimately $\tilde{Z}(3)$ needs to be removed from the list of all minimal cut sets. After that, the algorithm repeats the previous steps based on the assumption


```

# Algorithm for finding all (minimal) cut sets for the OD-pair (X,Y)

# Initialization

r = <number of (minimal) paths between X and Y>
P = <list of all (minimal) paths between X and Y (as lists themselves)>
Z = <empty list> # Already found (minimal) cut sets

# Recursive function for finding the next relevant edge of the candidate
# currentCut based on paths k,...,r
def findNextCutEdge(currentCut, k):

    # Search for next available path given that all links from the
    # candidate currentCut fail

    i = 1
    while i <= length(currentCut):
        if currentCut[i] in P[k]: # Path P[k] not available
            if k <= r-1: # Further paths to consider
                k = k+1 # Continue with checking path P[k+1]
                i = 0
            else: # No more paths to consider
                check(currentCut) # Check candidate currentCut
                return # Finish current recursive step
        i = i+1

    # Iteratively select new cut edge from current path and start next
    # recursive step

    for l = 1,...,length(P[k]):
        currentCut.append(P[k][l]) # Add element from P[k] to currentCut
        if k <= r-1: # Further paths to consider
            findNextCutEdge(currentCut, k+1)
            currentCut.pop() # Remove last element from currentCut
        else: # No more paths to consider
            check(currentCut) # Check candidate currentCut
            if l == length(P[k]): # No more elements to add from P[k]
                return # Finish current recursive step
            else:
                currentCut.pop() # Remove last element from currentCut

# Function for checking the cut set candidate as well as already found
# cut sets in Z with regard to minimality
def check(candidate):

    for j = 1,...,length(Z):
        if Z[j] ⊆ candidate: # candidate is not minimal or duplicate
            return # Reject candidate

    j = 1
    while j <= length(Z):
        if Z[j] ⊇ candidate: # Cut set Z[j] is not minimal
            delete Z[j] # Remove Z[j] from list of cut sets
            continue
        j = j+1
    Z.append(candidate) # Add candidate to the list Z

# Main function call
findNextCutEdge([], 1)

```

Figure 3: Recursive algorithm for finding all (minimal) cut sets between two given nodes of a network based on the knowledge of all (minimal) paths.

that link 11 in $P(1)$ is interrupted instead of link 10. By that, finally seven additional cut sets are found, four of them being not minimal. Moreover, also $\tilde{Z}(4)$ has to be rejected as a potential candidate in this context because of the smaller cut set $\tilde{Z}(7) := \{8, 11\}$.

In the end, the algorithm correctly provides the following minimal cut sets for the considered node E from the reduced network in Fig. 1b with regard to its connection to at least one of the supply nodes. Namely, $Z(1) := \{4, 10\}$, $Z(2) := \{8, 10, 12\}$, $Z(3) := \{9, 10\}$, $Z(4) := \{8, 11\}$, $Z(5) := \{4, 11, 12\}$ and $Z(6) := \{9, 11, 12\}$.

3 RESULTS

In the following, we present the analysis of the network from Fig. 1a with regard to the original questions from Section 1 based on the aforementioned methods. Figure 4 shows the derived probabilities R_{sys}^X for each regular node X from the network that this node remains connected to at least one of the given supply nodes depending on the link failure probability $p \in [0,1]$ that is assumed to be the same for all links.

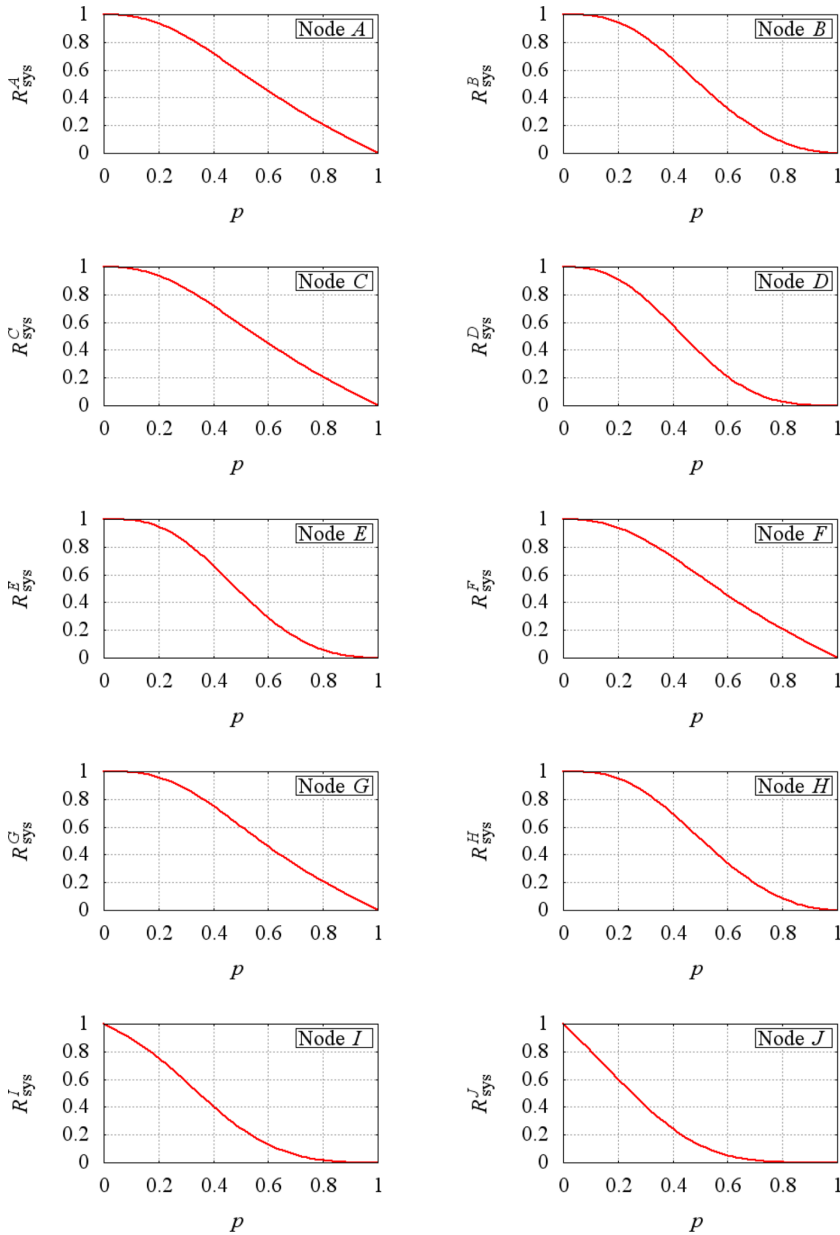


Figure 4: System reliability R_{sys}^X per node ($X = A, \dots, J$) depending on the link failure probability p in case of the exemplary network from Fig. 1a.

Obviously, the system reliability R_{sys}^X (and thus also the isolation risk $Q_{\text{sys}}^X := 1 - R_{\text{sys}}^X$) for each node $X \in \{A, \dots, J\}$ is a polynomial (but non-linear) function of p . Moreover, each node has its own characteristic curve in Fig. 4 reflecting the topological details of the network. Table 1 lists some concrete values of R_{sys}^X in case of $p = 1\%$, 5% and 10% . It turns out that whenever there are two or more completely disjoint paths between the considered (regular) node X and the existing supply nodes, the isolation risk usually (i.e., at least for small values of p) is much lower than p . This, in particular, explains why the nodes I (and J) are distinctly different from the others as there are no alternative paths in the network if link 13 (or link 14) fails.

The completely different behavior of nodes with and without alternative paths to the given supply nodes becomes most obvious in the theoretical networks from Fig. 5 which, so to say, are the extreme situations. Both networks are considered to equally consist of m links with

Table 1: Isolation risk Q_{sys}^X per node ($X = A, \dots, J$) in case of the exemplary network from Fig. 1a for some fixed values of p . The factor values $\lambda_{1\% \rightarrow 5\%}$ and $\lambda_{5\% \rightarrow 10\%}$ describe the rates of increase of Q_{sys}^X when raising p as denoted (i.e., $Q_{\text{sys}}^X[p = \alpha] \cdot \lambda_{\alpha \rightarrow \beta} = Q_{\text{sys}}^X[p = \beta]$).

Node	Isolation risk Q_{sys}^X				
	$p = 1\%$	$\lambda_{1\% \rightarrow 5\%}$	$p = 5\%$	$\lambda_{5\% \rightarrow 10\%}$	$p = 10\%$
A	0.0104%	28.7	0.2984%	4.6	1.3679%
B	0.0008%	124.1	0.0993%	7.8	0.7768%
C	0.0104%	28.7	0.2984%	4.6	1.3679%
D	0.0107%	31.6	0.3383%	5.0	1.6955%
E	0.0006%	136.0	0.0816%	8.4	0.6814%
F	0.0103%	28.0	0.2886%	4.5	1.3077%
G	0.0005%	132.6	0.0663%	8.2	0.5417%
H	0.0006%	132.2	0.0793%	8.2	0.6495%
I	1.0008%	5.1	5.0943%	2.1	10.6991%
J	1.9908%	4.9	9.8396%	2.0	19.6292%

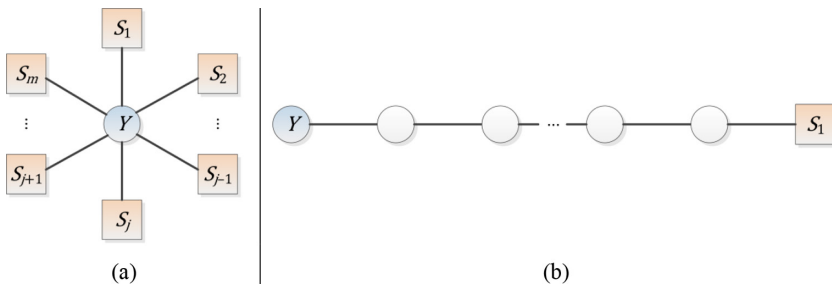


Figure 5: Theoretical networks with and without alternative paths between Y and the given supply nodes S_j ($j = 1, \dots, m$): (a) star topology; (b) line topology

the same link failure probability $p \in (0,1)$. Then, the modified graph (cf. Section 2.1) belonging to the star topology from Fig. 5a naturally consists of two nodes only that, however, are connected via m parallel links. That is, we obtain a multigraph in this case. On the contrary, the line network from Fig. 5b does not change at all in the modified graph, of course. The system reliability R_{sys} of the node Y with regard to its connection to (at least one of) the supply nodes is then given by (cf. Section 2)

$$R_{\text{sys}} = 1 - (1 - p)^m \xrightarrow{m \rightarrow \infty} 1 \quad (15)$$

in case of the star topology (see Fig. 5a) and

$$R_{\text{sys}} = p^m \xrightarrow{m \rightarrow \infty} 0 \quad (16)$$

for the line network (see Fig. 5b), respectively.

We finally ask for the relevance of each single link in context of the overall network topology (cf. Section 1). In the case of Figs 5a and b, it becomes easy to see that for symmetry reasons all links are equally relevant within each respective network. In the exemplary network from Fig. 1a, however, it is much more difficult to decide what the critical links are. As a kind of heuristic, Fig. 6a shows the total counts of how often each link appears in the cut sets of the nodes A, \dots, J . Assuming that links are more critical if they are part of many cut sets, it turns out that the links 3, 5 and 10 are the most important ones. On the contrary, the links 6 and 7 from the center of the network are seemingly much less critical. Clearly, the links 13 and 14 appear in very few cut sets (and thus are not very critical according to our simple definition of criticality) because their failure reduces the connectivity of the nodes I and J only whereas the majority of the network remains unaffected.

Needless to say, the simple counts from Fig. 6a are just a first heuristic for identifying critical links. A more detailed approach should also take account of the different sizes of the cut sets, for instance. This could be done by suitably weighting the link counts depending on the respective cut sets they are part of. In this context, note that the average size of a cut set in case of our example from Fig. 1a is 4.04. The complete size distribution is depicted in Fig. 6b.

4 CONCLUSION AND FURTHER DEVELOPMENT

The concept of terminal reliability is well-suited for analyzing the connectivity between two arbitrary nodes of a given network in case of stochastic link failures. The present contribution provides a simple method (graph modification, see Section 2.1) which enables us to also

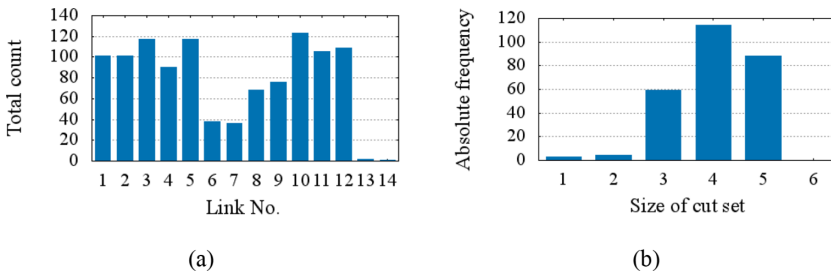


Figure 6: (a) Counts of link appearance in the minimal cut sets of all (regular) nodes from the exemplary network in Fig. 1a; (b) Size distribution of the cut sets.

consider the case where the destination is defined by a set of multiple nodes instead of a single node only. The most crucial point when applying terminal reliability, however, is the problem of scalability. For, besides the identification of all relevant paths and cut sets, the necessary expansion of the structure function ψ using Boolean algebra (cf. Eq. (6)) can become highly expensive computationally in case of large networks. In this context, clustering (see Section 2.2) is a basic step for reducing the efforts. Nonetheless, more advanced methods (cf. [6]) are required for large-scale applications.

Apart from that, further analyses should consider how to deal with multiple types of supply nodes as destination in parallel. That is, what are the system reliabilities per node when connections to more than one type of supply (e.g., fire department, hospital, police station) have to be available all the time? Moreover, we could ask for not only the isolation risk of single nodes but also how likely it is that M or more nodes lose their connection to the supply nodes at the same time where $M \geq 1$. Another interesting question would be: What happens if not only links but also supply nodes fail? How can this be modelled in context of the above theory and what are the effects on system reliability? The same, by the way, holds in case that we allow correlations with regard to the link failures instead of considering stochastically independent link indicator functions I_i only (cf. Section 1).

Finally, in Section 3 we shortly discussed a simple heuristic for identifying critical links based on the found cut sets. Of course, the presented approach needs to be enhanced in order to provide more distinguished results. Besides simple weighting as already mentioned (see Section 3), this could also include taking account of the potentially different importance among the nodes of the network. Highly populated areas or nodes with a high strategic relevance, for instance, could be considered as being more important than others in this context.

ACKNOWLEDGEMENTS

The authors acknowledge financial support by the German Helmholtz Association via institutional funding. The presented results are part of the VABENE++ project.

REFERENCES

- [1] Berdica, K., An introduction to road vulnerability: what has been done, is done and should be done. *Transport Policy*, **9**(2), pp. 117–127, 2002.
[https://doi.org/10.1016/S0967-070X\(02\)00011-2](https://doi.org/10.1016/S0967-070X(02)00011-2)
- [2] Bell, M.G.H., Kurauchi, F., Perera, S. & Wong, W., Investigating transport network vulnerability by capacity weighted spectral analysis. *Transportation Research Part B*, **99**, pp. 251–266, 2017.
<https://doi.org/10.1016/j.trb.2017.03.002>
- [3] TRB Resilience: Key Products & Projects – October 2017; Transportation Research Board (TRB), available at <http://onlinepubs.trb.org/Onlinepubs/dva/SecurityActivities.pdf> (accessed 3 January 2018).
- [4] Bell, M.G.H. & Iida, Y., *Transportation Network Analysis*, Wiley: Chichester, pp. 179–192, 1997.
- [5] Sedgewick, R. & Wayne, K., *Introduction to Programming in Java*, Addison Wesley: Boston, Massachusetts, 2017.
- [6] Wakabayashi, H. & Iida, Y., Upper and lower bounds of terminal reliability of road networks: an efficient method with Boolean algebra. *Journal of Natural Disaster Science*, **14**(1), pp. 29–44, 1992.